On the Construction of Minimally Redundant Reliable System Designs

By D. K. RAY-CHAUDHURI

(Manuscript received September 7, 1960)

Several authors have considered the possibility of increasing the reliability of large and complex binary digital systems by introducing some redundancy in the system. In a companion paper, Armstrong[1] proposes a scheme for applying error correction to a synchronous digital system. In this paper we develop a general mathematical theory for generating minimally redundant error-correcting codes for the scheme in question. This results in what are called "minimally redundant reliable systems." The problem of constructing minimally redundant reliable systems whose output is free of error when there is a fault in at most one block of the system is completely solved. An example is considered in detail showing how the mathematical theory can be actually applied.

I. INTRODUCTION

In complex binary digital systems employing a large number of blocks of electrical equipment it often is difficult to ensure a sufficient level of reliability of each single block of equipment. An attempt to attain the desired degree of reliability by improving the reliability of each block may prove to be uneconomical. On the other hand, by introducing some redundancy in the system, it is possible to construct highly reliable complex systems, even though each single block is not as highly reliable. Moore and Shannon,[2] Tryon,[3] Von Neumann,[4] Lofgren[5] and Armstrong[1] have considered the problem of constructing reliable system designs. In this paper a general mathematical theory has been developed for the construction of minimally redundant reliable system designs, based on the scheme outlined by Armstrong.[1] This theory is closely related to the theory of error-correcting codes. The problem of constructing minimally redundant system designs whose outputs will be free of error whenever there is fault in at most one block of the system is completely solved in this paper.

II. FORMULATION OF THE PROBLEM

Suppose there are m binary input variables $X_1, X_2, \cdots, X_m$. Let B_m denote the set of 2^m m-place binary sequences. Every set of values of the m binary input variables will be regarded as an element of B_m. Any mapping of B_m into B_1 will be called a *Boolean function* of the m input variables $X_1, X_2, \cdots, X_m$. For the sake of brevity, the collection of m input variables will be denoted by X. Let

$$f_{11}, f_{12}, \cdots, f_{1p}, \qquad f_{21}, f_{22}, \cdots, f_{2p}, \qquad f_{k1}, f_{k2}, \cdots, f_{kp}$$

be pk Boolean functions of the m binary variables $X_1, X_2, \cdots, X_m$. Our problem is to construct a system which will synthesize the pk Boolean functions with a high degree of reliability. The system uses blocks of electrical equipments each of which can synthesize p Boolean functions. For the sake of brevity, a collection of p Boolean functions, will be called a *Boolean p-function*. Thus $f_i = (f_{i1}, f_{i2}, \cdots, f_{ip})$ is a Boolean p-function. Any Boolean p-function is a mapping of B_m into B_p. Each block of our system synthesizes a Boolean p-function. Fig. 1 is a schematic diagram for the original nonredundant system.

The blocks act as units in the system. If there is a fault in a block, then some or all the p outputs of the block are erroneous. In other words, in the case of a fault a block will synthesize the corresponding Boolean p-function wrongly. Let V_p^1 denote the set of 2^p binary p-tuples. Then any Boolean p-function takes values on V_p^1. Let V_p^k denote the set of k-vectors $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_k)$, where each α_i is an element of V_p^1, $i = 1, 2 \cdots, k$. Let $f = (f_1, f_2, \cdots, f_k)$. Then f can be regarded as a mapping of B_m onto V_p^k. We shall define the addition of p-tuples as the usual mod 2 addition. For example, if $p = 3$, $\alpha_1 = (001)$ and $\alpha_2 = (101)$, then $\alpha_1 + \alpha_2 = (100)$. Let α and α' be two elements of V_p^k given by $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_k)$ and $\alpha' = (\alpha_1', \alpha_2', \cdots, \alpha_k')$. The sum $\alpha + \alpha'$

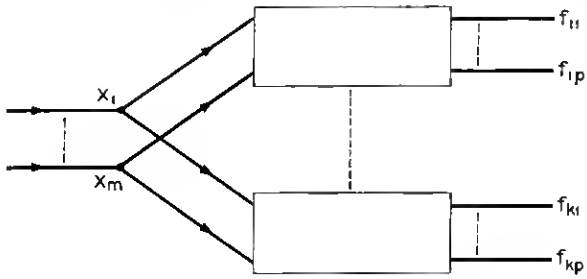

Fig. 1 — Original nonredundant system.

is defined to be the element $(\alpha_1 + \alpha_1', \cdots, \alpha_k + \alpha_k')$. The p-tuple $(00 \cdots 0)$ will be called the *null element* of V_p^1. The weight $\omega(\alpha)$ of the k-vector α is defined to be the number of nonnull elements among $\alpha_1, \alpha_2, \cdots, \alpha_k$. For any particular value X' of the input variables $f(X')$ is a vector in V_p^k. Suppose there are faults in t $(t < k)$ blocks. Then t of the functions $f_1, f_2, \cdots, f_k$ will be synthesized wrongly. Hence the output will be the vector $f(X') + \epsilon$, where $\epsilon = (\epsilon_1, \epsilon_2, \cdots, \epsilon_k)$ is a vector in V_p^k with weight t. While designing a system to synthesize the Boolean function f, one might require that whenever the number of faulty blocks is t or less, the output is error-free. One can achieve this by introducing some redundancy in the system, i.e., by synthesizing $(k + r)$ Boolean p-functions and adding a logical corrector unit to the system.

Suppose $\varphi_1, \varphi_2, \cdots, \varphi_n$ are n Boolean p-functions and C is a mapping of V_p^n onto V_p^k. We shall consider $\varphi = (\varphi_1, \varphi_2, \cdots, \varphi_n)$ as a function from B_m to V_p^n. For every value X' of X, $\varphi(X)$ is an element of V_p^n. Suppose the functions φ and C possess the property P stated below:

For every vector ϵ belonging to V_p^n with $\omega(\epsilon)$ not exceeding t and every value X' of the input variable X,

$$C(\varphi(X') + \epsilon) = f(X'). \tag{1}$$

The functions φ and C enable us to construct a system which will synthesize the k Boolean p-functions $f_1, f_2, \cdots, f_k$ free of error whenever the number of faulty blocks in the system is t or less. The n Boolean p-functions $\varphi_1, \varphi_2, \cdots, \varphi_n$ can be considered as a collection of np Boolean functions of m input variables. Therefore we can easily obtain the logical design of a system which will synthesize these np Boolean functions. This system will be called the *encoder subsystem*. Similarly, the function C can be considered as a collection of pk Boolean functions of np binary input variables, and therefore we can obtain a system which will synthesize these pk functions. This system will be called the *corrector subsystem*. The np outputs of the encoder subsystem will be the inputs of the corrector subsystem. Now it is easily seen that, because of the property P of the functions φ and C, whenever the number of faulty blocks in the encoder subsystem is t or less and the corrector unit is free of error, the pk outputs of the corrector subsystem will be

$$\begin{aligned} f(X) &= \{f_1(X), f_2(X), \cdots, f_k(X)\} \\ &= \{f_{11}(X), f_{12}(X), \cdots, f_{1p}(X), f_{21}(X), f_{22}(X), \cdots, f_{2p}(X), \cdots, \\ &\quad f_{k1}(X), f_{k2}(X), \cdots, f_{kp}(X)\}. \end{aligned} \tag{2}$$

A schematic diagram for the whole system is given in Fig. 2. In view of

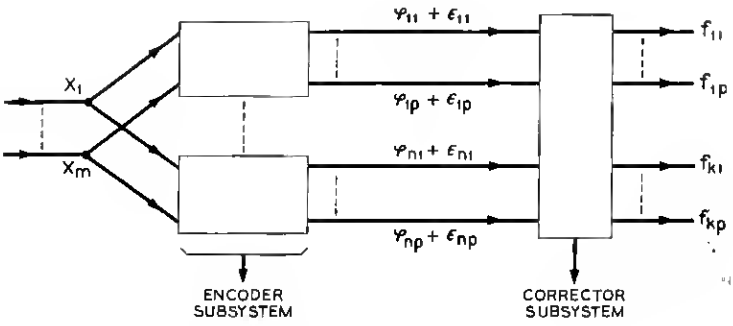

Fig. 2 — Whole system.

the above discussion, the following two definitions given below are meaningful.

Definition 1: The functions $\varphi_1, \varphi_2, \cdots, \varphi_n$ and C possessing the property P stated in (1) will be called a *reliable system design* of order t and redundancy $r = n - k$ for the k Boolean p-functions $f_1, f_2, \cdots, f_k$.

Definition 2: A reliable system design of order t and redundancy r_0 for the k Boolean p-functions $f_1, f_2, \cdots, f_k$ will be called *minimally redundant* if the redundancy r of any other reliable system design of order t for the same functions is not less than r_0.

In the present paper we have given a method of obtaining a minimally redundant system design of order 1 for any set of k Boolean p-functions for arbitrary k and p. System designs of higher order will be given in a subsequent paper.

We have used the redundancy r as a measure of the extra amount of equipment which has to be used for making the system reliable. And hence we seek the system which has minimum possible value of the redundancy r. It should be pointed out that we assumed that the corrector subsystem does not make any error at all. Therefore, to make the whole development practically feasible, it is imperative that either the amount of equipment necessary for the corrector subsystem is small in comparison to the amount of equipment necessary for the whole system, or that other steps be taken, such as are suggested in Ref. 1, to ensure reliability of the corrector system. We have not used any mathematical criterion to incorporate this requirement in the development of the theory.

III. LOWER BOUNDS ON THE REDUNDANCY r OF A RELIABLE SYSTEM DESIGN OF ORDER t

Consider two vectors α and α' belonging to V_p^n. The distance $d(\alpha,\alpha')$ between these two elements of V_p^n is defined to be $\omega(\alpha + \alpha')$. Thus if

$\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)$ and $\alpha' = (\alpha_1', \alpha_2', \cdots, \alpha_n')$, then $d(\alpha,\alpha')$ is equal to the number of integers i for which $\alpha_i \neq \alpha_i'$, $i = 1,2,\cdots,n$. For example, if $p = 2$, $n = 3$, and $\alpha = (01,11,10)$ and $\alpha' = (01,10,00)$, then $\alpha + \alpha' = (00,01,10)$ and $d(\alpha,\alpha') = 2$. It can be easily checked that the distance defined above satisfies the three conditions of a distance function. We have seen that Boolean p-functions as defined in Section II can be considered as mappings of B_m into V_p^1. A Boolean p-function f_1 will be called a *nondegenerate* Boolean p-function if for any element α_1 of V_p^1, there is a value X' of the input variable X for which $f_1(X') = \alpha_1$. We shall assume that all Boolean p-functions appearing in our discussion are nondegenerate. In the following we have $s = 2^p$ and $n = k + r$.

Theorem 1: A necessary and sufficient condition that there exists a reliable system design of order t and redundancy r for the k Boolean p-functions $f_1, f_2, \cdots, f_k$ is that there exists a subset A of V_p^n containing s^k elements such that $d(\alpha,\alpha') \geqq 2t + 1$; $\alpha,\alpha' \in A'$, $\alpha \neq \alpha'$.

Proof: Necessity. Suppose there exists a reliable system design of order t. Let the encoder functions be $\varphi = (\varphi_1,\varphi_2,\cdots,\varphi_n)$ and the corrector function be C. For every value X' of the input variable X, $\varphi(X')$ is a vector of V_p^n. Consider the set

$$A = \{\varphi(X') \mid X' \in B_m\}.$$

Using the fact that the Boolean p-functions $f_1, f_2, \cdots, f_k$ are nondegenerate functions, it follows easily that the set A contains at least s^k vectors of V_p^n. Consider two distinct vectors α and α' of the set A. If possible, suppose $d(\alpha,\alpha') \leqq 2t$. Since $d(\alpha,\alpha') \leqq 2t$, we can find a vector ϵ of V_p^n such that $\alpha + \epsilon = \alpha' + \epsilon$ and $\omega(\epsilon) \leqq t$. Since $\omega(\epsilon) \leqq t$, we have

$$C(\alpha + \epsilon) = C(\alpha' + \epsilon) = \alpha = \alpha'. \tag{3}$$

Equation (3) contradicts the assumption that α and α' are distinct vectors of A. This completes the proof of necessity.

Sufficiency. Suppose A is a subset of V_p^n containing s^k elements and having the property that $d(\alpha,\alpha') \geqq 2t + 1$; $\alpha,\alpha' \in A$, $\alpha \neq \alpha'$. We set up a one-to-one correspondence between the s^k vectors of V_p^k and the s^k vectors of A. For every value X' of the input variables X, $f(X') = [f_1(X'),f_2(X'),\cdots,f_k(X')]$ is a vector of V_p^k and there is a corresponding vector α of V_p^n belonging to A. The encoder function $\varphi = (\varphi_1, \varphi_2, \cdots, \varphi_n)$ is defined by

$$\begin{aligned} \varphi(X') &= [\varphi_1(X'),\varphi_2(X'),\cdots,\varphi_n(X')] \\ &= (\alpha_1, \alpha_2, \cdots, \alpha_n) \\ &= \alpha, \end{aligned} \tag{4}$$

where α is the vector of V_p^n belonging to A and corresponding to the vector $f(X')$ of V_p^k. The corrector function C is defined in the following manner. Let $\gamma = (\gamma_1, \gamma_2, \cdots, \gamma_n)$ be an arbitrary vector of V_p^n. First we choose a vector α belonging to A such that $d(\gamma,\alpha) \leqq d(\gamma,\alpha')$, $\alpha,\alpha' \in A$. Let $\beta = (\beta_1, \beta_2, \cdots, \beta_k)$ be the vector of V_p^k which corresponds to α. Then we define

$$C(\gamma) = \beta. \tag{5}$$

Thus C is a mapping of V_p^n onto V_p^k. It is easy to check that the encoder function φ and the corrector function C defined above possess the property P stated in Section II. This completes the proof of sufficiency.

Theorem 2: If there exists a reliable system design of order t and redundancy r for k Boolean p-functions, then

$$s^r \geqq 1 + \binom{n}{1}(s-1) + \binom{n}{2}(s-1)^2 + \cdots + \binom{n}{t}(s-1)^t, \tag{6}$$

where $n = k + r$ and $s = 2^p$.

Proof: From Theorem 1, it is necessary that there exists a subset A of V_p^n with the property that

$$d(\alpha,\alpha') \geqq 2t + 1; \qquad \alpha,\alpha' \in A, \quad \alpha \neq \alpha'. \tag{7}$$

Let S_α denote the set of vectors γ of V_p^n with the property that $d(\gamma,\alpha) \leqq t$. It follows easily from (7) that, for any two distinct vectors α and α' of A, the sets S_α and $S_{\alpha'}$ do not have any common element. Let $S_\alpha^{(k)}$ denote the set of elements of V_p^n which have distance k from α, $k = 0,1,2,\cdots,t$. Obviously S_α is the union of the $(t+1)$ sets $S_\alpha^{(k)}$; $k = 0,1,\cdots,t$. $S_\alpha^{(k)}$ contains

$$\binom{n}{k}(s-1)^k$$

elements. Hence S_α contains

$$1 + \binom{n}{1}(s-1) + \binom{n}{2}(s-1)^2 + \cdots + \binom{n}{t}(s-1)^t$$

elements. There are s^k such nonoverlapping sets and the total number of elements of V_p^n is s^n. Hence we have

$$s^n \geqq s^k\left[1 + \binom{n}{1}(s-1) + \binom{n}{2}(s-1)^2 + \cdots + \binom{n}{t}(s-1)^t\right]. \tag{8}$$

Theorem 2 follows from (8).

Theorem 2 gives a lower bound on the redundancy r of a reliable

system design of order t for k Boolean p-functions. Theorem 2 is actually a generalization of a result of Hamming.[6]

Let $n_t(r)$ denote the maximum integer n for which there exists a reliable system design of order t and redundancy r for $k = n - r$ Boolean p-functions. For $t = 1$, the inequality (6) becomes

$$s^r \leqq \binom{n}{1} (s - 1).$$

Hence we have

$$n_1(r) \leqq \frac{s^r - 1}{s - 1}.$$

In Section V we shall show that

$$n_1(r) = \frac{s^r - 1}{s - 1}.$$

If there exists a reliable system design of order t and redundancy r for k Boolean p-functions, then $n_t(r) \geqq k + r$.

Lemma 1:

$$n_t(r + 1) \geqq n_t(r) + 1.$$

Proof: Suppose $n_t(r) = n$. Then there exists a reliable system design of order t and redundancy r for $k = n - r$ Boolean p-functions. Hence by Theorem 1 there exists a subset A of ${V_p}^n$ containing s^k elements with the property that $d(\alpha,\alpha') \geqq 2t + 1$; $\alpha,\alpha' \in A$, $\alpha \neq \alpha'$. To every vector $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)$, we associate the vector

$$\bar{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_n, 0)$$

of ${V_p}^{n+1}$. Thus we have a subset $\bar{A}$ of ${V_p}^{n+1}$ containing s^k elements and also possessing the property that $d(\bar{\alpha},\bar{\alpha}') \geqq 2t + 1$; $\bar{\alpha},\bar{\alpha}' \in \bar{A}$, $\bar{\alpha} \neq \bar{\alpha}'$. Hence, by Theorem 1, we can obtain a reliable system design of order t and redundancy $n + 1 - k = r + 1$. It follows that

$$n_t(r + 1) \geqq k + r + 1 = n + 1 = n_t(r) + 1.$$

Theorem 3: If for a reliable system design of order t and redundancy r for k Boolean p-functions we have

$$n_t(r - 1) - (r - 1) < k \leqq n_t(r) - r, \tag{9}$$

then the design is minimally redundant.

Proof: If possible, suppose the system is not minimally redundant. Then there exists a reliable system design of order t and redundancy

$r - c$ for k Boolean p-functions where c is some positive integer. Then we have $n_t(r - c) \geqq k + (r - c)$. By Lemma 1,

$$\begin{aligned} n_t(r - 1) &\geqq n_t(r - c) + (c - 1) \\ &\geqq k + (r - 1). \end{aligned} \tag{10}$$

The inequality (10) contradicts the inequality (9); hence the theorem is established.

IV. LINEAR SYSTEM DESIGNS

In this section we shall consider a particular subclass of system designs called the *linear system designs*. To define the linear system designs, we have to use the theory of finite fields. Let K be the finite field of characteristic 2 containing $s = 2^p$ elements and x denote a primitive element of K. Any binary p-tuple $(a_0, a_1, \cdots, a_{p-1})$ will be made to correspond to the element $a_0 + a_1x + \cdots + a_{p-1}x^{p-1}$ of K and vice versa. An element $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)$ of ${V_p}^n$ now will be considered as an n-vector with elements in K. The weight $\omega(\alpha)$ of α is equal to the number of nonnull elements among $\alpha_1, \alpha_2, \cdots, \alpha_n$. The sum of two vectors $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_n)$ and $\alpha' = (\alpha_1', \alpha_2', \cdots, \alpha_n')$ is defined to be

$$\alpha + \alpha' = (\alpha_1 + \alpha_1', \alpha_2 + \alpha_2', \cdots, \alpha_n + \alpha_n').$$

Obviously ${V_p}^n$ is a vector space over K. Consider a system design for k Boolean p-functions. Suppose the encoder function is

$$\varphi = (\varphi_1, \varphi_2, \cdots, \varphi_n).$$

For every value X' of the input variable X,

$$\varphi(X') = [\varphi_1(X'), \varphi_2(X'), \cdots, \varphi_n(X')]$$

is a vector belonging to ${V_p}^n$. Let

$$A = \{\varphi(X') \mid X' \in B_m\}. \tag{11}$$

Definition 3: A system design for k Boolean p-functions is said to be a *linear system design* if the subset A of ${V_p}^n$ defined by (11) is a vector space over K.

Lemma 2: A necessary and sufficient condition that a reliable linear system design for k Boolean p-functions is of order t is that the weight of any nonnull vector of the set A defined in (11) is not less than $(2t + 1)$.

Proof: Because of Theorem 1, it would be sufficient to show that

$$d(\alpha,\alpha') \geqq 2t + 1; \qquad \alpha,\alpha' \in A, \quad \alpha \neq \alpha'. \tag{12}$$

By definition $d(\alpha,\alpha') = \omega(\alpha + \alpha')$. Since A is a vector space, $\alpha + \alpha'$ is an element of A and also $\alpha + \alpha'$ is a nonnull element of A. Hence it follows that (12) will hold if and only if $\omega(\alpha) \geq 2t + 1$ for every nonnull element α of A.

Definition 4: A matrix M with elements in K will be said to have the (P_t)-property if no t rows of the matrix are linearly dependent.

Theorem 4: A necessary and sufficient condition for the existence of a reliable linear system design of order t and redundancy r for k Boolean p-functions is that there exists a matrix M with $n = (k + r)$ rows and r columns with elements in K which possesses (P_{2t})-property.

Proof: Sufficiency. Suppose the matrix M is given by

$$M = \begin{bmatrix} m_{11} & m_{12} \cdots m_{1r} \\ m_{21} & m_{22} \cdots m_{2r} \\ \vdots & \vdots \quad \vdots \\ m_{n1} & m_{n2} \quad m_{nr} \end{bmatrix}. \tag{13}$$

Let A denote the vector space orthogonal to the vector space generated by the r column vectors of M. A contains at least s^k elements. It would be sufficient to show that the weight of any nonnull vector α of A is at least $(2t + 1)$. If possible, suppose A contains a nonnull vector with weight less than $(2t + 1)$. For simplicity of writing assume that the vector $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_{2t}, 0, \cdots, 0)$ belongs to A where $\alpha_1, \alpha_2, \cdots, \alpha_{2t}$ are nonzero elements of K. Then we have

$$\alpha_1 m_{1i} + \alpha_2 m_{2i} + \cdots + \alpha_{2t} m_{2ti} = 0; \qquad i = 1,2,\cdots,r. \tag{14}$$

Equation (14) implies that the first $2t$ vectors of the matrix M are linearly dependent which is a contradiction. This completes the proof of sufficiency. Necessity can be proved by exactly similar arguments.

The reader acquainted with the literature on error-correcting linear codes would recognize from Theorem 4 that a reliable linear system design of order t for k Boolean p-functions exists if and only if a t-error correcting linear code in $s(= 2^p)$ symbols with n places and k information places exists. Lemma 1 and Theorem 4 given above are not new results; they were proved by Bose[7] and Zierler[8] in a different form. We have included short proofs for these results for the sake of completeness.

V. MINIMALLY REDUNDANT LINEAR SYSTEM DESIGNS OF ORDER 1

In this section we shall give methods for constructing minimally redundant linear system designs of order 1 for k Boolean p-functions for any arbitrary value of k and p.

Theorem 5:

$$n_1(r) = \frac{s^r - 1}{s - 1}.$$

Proof: In Section III we proved that

$$n_1(r) \leqq \frac{s^r - 1}{s - 1}.$$

Hence it would be sufficient to show that

$$n_1(r) \geqq \frac{s^r - 1}{s - 1}. \tag{15}$$

To prove (15) we shall construct a matrix M with r columns and $n = (s^r - 1)/(s - 1)$ rows which has (P_2)-property. We shall denote the elements of K by $0,1,\alpha_2, \cdots, \alpha_{s-1}$, where 0 is the null element and 1 is the multiplicative identity. Consider the matrix M given by

$$M = \begin{bmatrix} M_1 \\ I_r \end{bmatrix}, \tag{16}$$

where I_r is the identity matrix with r rows and r columns and M_1 is a matrix with r columns and k $(= n - r)$ rows given below:

$$M_1 = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & 1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 1 & \alpha_{s-1} \\ 0 & 0 & \cdots & 1 & 1 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & \alpha_{s-1} & \alpha_{s-1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 1 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ 1 & \alpha_{s-1} & & \alpha_{s-1} & \alpha_{s-1} & \alpha_{s-1} \end{bmatrix}. \tag{17}$$

It can be easily checked that the matrix M has (P_2)-property; i.e., no two rows of M are linearly dependent. This completes the proof of Theorem 5.

It should be observed that Theorem 5 enables us to construct minimally redundant system designs of order 1 for any arbitrary values of k and p. For given k, we find out the integer r for which

$$n_1(r - 1) - (r - 1) < k \leqq n_1(r) - r.$$

If $n_1(r) - r = k$, we construct the matrix M with r columns and $n_1(r)$ rows as defined in (16) and then obtain the system design as illustrated

in the proof of Theorem 4. If $n_1(r) - r > k$, then we delete $n_1(r) - (k + r)$ rows from M_1 and thus obtain a matrix M with (P_2)-property which has r columns and $(k + r)$ rows. From Theorem 3, is follows that the resulting system design will be minimally redundant. Now we shall give explicitly the encoder function of the minimally redundant system designs of order 1. Let

$$\underset{(n\times k)}{B} = \begin{bmatrix} I_k \\ M_1' \end{bmatrix},$$

where I_k is the identity matrix with k rows and k columns and M_1' is the transpose of the matrix M_1. It can be verified that the k column vectors of B are orthogonal to each of the r column vectors of M. The k column vectors of B generate the vector space A and every nonnull vector of A has weight greater than $2t$. For the sake of convenience of description, we write

$$M_1' = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1k} \\ \vdots & \vdots & & \vdots \\ m_{r1} & m_{r2} & \cdots & m_{rk} \end{bmatrix}.$$

To define the encoder function φ, we must set up a one-to-one correspondence between the s^k vectors of $V_p^{\,k}$ and the s^k vectors of A. We make the vector $(\alpha_1, \alpha_2, \cdots, \alpha_k)$ of $V_p^{\,k}$ correspond to the vector $(\alpha_1, \alpha_2, \cdots, \alpha_k, \alpha_{k+1}, \cdots, \alpha_n)$ of A, where

$$\begin{aligned} \alpha_{k+1} &= \alpha_1 m_{11} + \alpha_2 m_{12} + \cdots + \alpha_k m_{1k} \\ &\vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \\ \alpha_n &= \alpha_1 m_{r1} + \alpha_2 m_{r2} + \cdots + \alpha_k m_{rk}. \end{aligned}$$

Hence, if

$$\begin{aligned} f(X') &= [f_1(X'), f_2(X'), \cdots, f_k(X')] \\ &= (\alpha_1, \alpha_2, \cdots, \alpha_k), \\ \varphi(X') &= [\varphi_1(X'), \varphi_2(X'), \cdots, \varphi_k(X'), \varphi_{k+1}(X'), \cdots, \varphi_n(X')] \\ &= (\alpha_1, \alpha_2, \cdots, \alpha_k, \alpha_{k+1}, \cdots, \alpha_n). \end{aligned}$$

Therefore it follows that we have

$$\begin{aligned} \varphi_1(X) &= f_1(X) \\ &\vdots \qquad \vdots \\ \varphi_k(X) &= f_k(X), \\ \varphi_{k+1}(X) &= m_{11} f_1(X) + m_{12} f_2(X) + \cdots + m_{1k} f_k(X) \\ &\vdots \qquad\quad \vdots \qquad\quad \vdots \qquad\quad \vdots \\ \varphi_n(X) &= m_{r1} f_1(X) + m_{r2} f_2(X) + \cdots + m_{rk} f_k(X). \end{aligned} \tag{18}$$

It should be pointed out that in (18), for any particular value X' of X, $f_k(X')$ and m_{ji} ($j = 1,2,\cdots,r$; $i = 1,2,\cdots,k$) are elements of K and the operations of addition and multiplication are as in K. The corrector function can be built up, from the vector space A by the method already described in connection with the proof of the sufficiency part of Theorem 1. In the following, we shall give an alternative simpler method of building up a corrector subsystem which will correct one error in the encoder subsystem.

Suppose the output of the encoder subsystem is the vector $\gamma = (\gamma_1, \gamma_2, \cdots, \gamma_n)$ of ${V_p}^n$. If X' is the value of the input variable, then $\gamma = \varphi(X') + \epsilon$ where $\varphi(X') = (\varphi_1(X'), \varphi_2(X'), \cdots, \varphi_n(X'))$ and $\epsilon = (\epsilon_1, \epsilon_2, \cdots, \epsilon_n)$ is the error vector. From our method of constructing the encoder function, we have

$$\underset{(1\times n)}{\varphi(X')}\ \underset{(n\times r)}{M} = \underset{(1\times r)}{0}.$$

Hence

$$\gamma M = \epsilon M. \tag{19}$$

Let

$$\epsilon M = \delta = (\delta_1, \delta_2, \cdots, \delta_n).$$

Then

$$\begin{aligned} \delta_1 &= \gamma_{k+1} + m_{11}\gamma_1 + \cdots + m_{1k}\gamma_k \\ &\vdots \quad\quad \vdots \quad\quad\quad \vdots \\ \delta_r &= \gamma_n + m_{r1}\gamma_1 + \cdots + m_{rk}\gamma_k . \end{aligned} \tag{20}$$

If there is error in one block of the encoder subsystem, ϵ will have one nonzero element among its coordinates. Suppose the lth coordinate of ϵ is a nonzero element λ of K. Then $\epsilon = (0,0,\cdots,\lambda,0,\cdots,0)$, $1 \leqq l \leqq n$. Let us denote the lth row vector of the matrix M by

$$R_l = (c_{l1}, c_{l2}, \cdots, c_{lr}).$$

Then the following equations hold:

$$\begin{aligned} \delta &= \epsilon M = \lambda R_l \\ \delta_j &= \lambda c_{lj}, \qquad j = 1,2,\cdots,r. \end{aligned} \tag{21}$$

Conversely, if for a given output vector γ the vector δ computed by (20) satisfies (21) and there is one block of the encoder subsystem in error, then the error vector ϵ will have λ as its lth coordinate and zero as the other coordinates.

From the above discussion, it is clear that a corrector subsystem which

observes the rules given below will accomplish the job of correcting errors in one block of the encoder subsystem and produce the k Boolean p-function $f_1(X), f_2(X), \cdots, f_k(X)$ as its output. The rules are

i. Compute δ as defined in (20).

ii. If δ is the null vector, the k outputs would be given by $\beta_i = \gamma_i$, $i = 1,2,\cdots,k$.

iii. If δ is not the null vector, find out the integer l for which the vector λR_l for some $\lambda \in K$ has maximum number of common coordinates with δ. If $l > k$, the k outputs are $\beta_i = \gamma_i$, $i = 1,2,\cdots,k$. If $l < k$, the k outputs are $\beta_1 = \gamma_i$, $\beta_2 = \gamma_2$, $\cdots$, $\beta_l = \gamma_l + \lambda, \cdots, \beta_k = \gamma_k$.

VI. AN EXAMPLE

In this section we shall give an example to show how the theory developed in this paper can be applied.

Suppose $m = 3$, $p = 2$, $k = 3$ and $t = 1$. From Section V, we can see that for the minimally redundant system $r = 2$. Suppose the three Boolean two-functions to be synthesized are

$$\begin{aligned} f_1(X) &= [f_{11}(X), f_{12}(X)] \\ &= (X_1 \cdot X_2, X_1 \oplus X_2), \\ f_2(X) &= [f_{21}(X), f_{22}(X)] \\ &= (X_2 \cdot X_3, X_2 \oplus X_3), \\ f_3(X) &= [f_{31}(X), f_{32}(X)] \\ &= (X_1 \cdot X_3, X_1 \oplus X_3) \end{aligned}$$

where the symbols $\oplus$ and $\cdot$ are respectively used to denote the Boolean operations of additions (OR) and multiplication (AND) between two binary variables. Let k denote the field containing four elements. Let t be a primitive element of the field. The polynomial $t^2 + t + 1$ is a minimum function and every element of the field satisfies the equation $x^3 = 1$. The four elements are shown below in terms of the primitive element t, and their correspondence with binary 2-vectors is also pointed out:

$$\begin{aligned} \alpha_0 &= 0 = 0 + 0t \leftrightarrow (0,0), \\ \alpha_1 &= 1 = 1 + 0t \leftrightarrow (1,0), \\ \alpha_2 &= t = 0 + 1t \leftrightarrow (0,1), \\ \alpha_3 &= t^2 = 1 + 1t \leftrightarrow (1,1). \end{aligned}$$

In view of the correspondence between the hinary 2-vectors and the elements of K, any particular value of a Boolean 2-function will be considered as an element of K. For example, if

$$f_1(X') = (1,1),$$

then

$$f_1(X') = \alpha_3 .$$

Addition and multiplication between the elements of K are shown in the tables given below:

Addition Table

	α_0	α_1	α_2	α_3
α_0	α_0	α_1	α_2	α_3
α_1	α_1	α_0	α_3	α_2
α_2	α_2	α_3	α_0	α_1
α_3	α_3	α_2	α_1	α_0

Multiplication Table

	α_0	α_1	α_2	α_3
α_0	α_0	α_0	α_0	α_0
α_1	α_0	α_1	α_2	α_3
α_2	α_0	α_2	α_3	α_1
α_3	α_0	α_3	α_1	α_2

The sum of two elements α_i and α_j is obtained by adding the corresponding polynomials in t modulo 2 ($i, j = 0, 1, 2, 3$). The product of two elements is obtained by multiplying the corresponding polynomials modulo 2 and modulo $(t^2 + t + 1)$. From Section V we have $m_{11} = m_{12} = m_{13} = \alpha_1$ and $m_{21} = \alpha_1$, $m_{22} = \alpha_2$ and $m_{23} = \alpha_3$. Therefore the five encoder Boolean 2-functions are given by

$$\varphi_i(X') = f_i(X'), \qquad i = 1, 2, 3,$$

$$\varphi_4(X') = f_1(X') + f_2(X') + f_3(X')$$

and

$$\varphi_5(X') = f_1(X') + \alpha_2 f_2(X') + \alpha_3 f_3(X').$$

Hence, $\varphi_{41}(X) = (X_1 \cdot X_2) + (X_2 \cdot X_3) + (X_1 \cdot X_3)$ and $\varphi_{42}(X) = (X_1 \oplus X_2) + (X_2 \oplus X_3) + (X_3 \oplus X_1)$, where $+$, $\oplus$ and $\cdot$ respectively denote mod 2 addition, Boolean addition (OR) and Boolean multiplication (AND) between two binary variables.

The truth table for the two Boolean functions φ_{51} and φ_{52} is given below:

X_1	X_2	X_3	φ_{51}	φ_{52}
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	1	1
1	1	0	1	0
1	1	1	0	0

The computation of this table will be illustrated by one example. Suppose $X_1 = 0$, $X_2 = 1$, $X_3 = 1$. Then $f_1(X) = (0,1) = \alpha_2$, $f_2(X) = (1,1) = \alpha_3$ and $f_3(X) = (0,1) = \alpha_2$. So $\varphi_5(X) = \alpha_2 + \alpha_2\alpha_3 + \alpha_3\alpha_2 = \alpha_2 = (0,1)$. And so $\varphi_{51}(X) = 0$ and $\varphi_{52}(X) = 1$. The corrector subsystem uses the outputs $\gamma_i = (\gamma_{i1},\gamma_{i2})$, $i = 1, 2, 3, 4, 5$, of the encoder subsystem as inputs. The final outputs $\beta_i = (\beta_{i1}, \beta_{i2})$, $i = 1, 2, 3$ of the corrector subsystem will be built up in several stages. From Section V, $\delta_1 = \gamma_4 + \gamma_1 + \gamma_2 + \gamma_3$ and $\delta_2 = \gamma_5 + \gamma_1 + \alpha_2\gamma_2 + \alpha_3\gamma_3$. At the first stage the corrector subsystem synthesizes $\eta_i = \alpha_i\gamma_i$, $i = 2, 3$. The truth tables for (η_{i1}, η_{i2}), $i = 2, 3$ are given below:

γ_{21}	γ_{22}	η_{21}	η_{22}	γ_{31}	γ_{32}	η_{31}	η_{32}
0	0	0	0	0	0	0	0
0	1	1	1	0	1	1	0
1	0	0	1	1	0	1	1
1	1	1	0	1	1	0	1

At the second stage, the binary 2-tuples δ_1, and δ_2 are synthesized. We have

$$\begin{aligned} \delta_{11} &= \gamma_{41} + \gamma_{11} + \gamma_{21} + \gamma_{31}, \\ \delta_{21} &= \gamma_{42} + \gamma_{21} + \gamma_{22} + \gamma_{32}, \\ \delta_{21} &= \gamma_{51} + \gamma_{11} + \eta_{21} + \eta_{31}, \\ \delta_{22} &= \gamma_{52} + \gamma_{12} + \eta_{22} + \eta_{32}. \end{aligned} \tag{22}$$

The addition between the binary variables in (22) is modulo 2 addition. At the third stage, the three binary 2-tuples, ϵ_1, ϵ_2 and ϵ_3, which are the first three coordinates of the error vector ϵ are synthesized as Boolean functions of the δ's. The part of the truth table in which at least one of the ϵ's takes the value 1 is given below:

δ_{11}	δ_{12}	δ_{21}	δ_{22}	ϵ_{11}	ϵ_{12}	ϵ_{21}	ϵ_{22}	ϵ_{31}	ϵ_{32}
0	1	0	1	0	1	0	0	0	0
0	1	1	0	0	0	0	0	0	1
0	1	1	1	0	0	0	1	0	0
1	0	0	1	0	0	1	0	0	0
1	0	1	0	1	0	0	0	0	0
1	0	1	1	0	0	0	0	1	0
1	1	0	1	0	0	0	0	1	1
1	1	1	0	0	0	1	1	0	0
1	1	1	1	1	1	0	0	0	0

The truth table given below is computed from the rules given in Section V. In the case of our example,

$$M = \begin{bmatrix} \alpha_1 & \alpha_1 \\ \alpha_1 & \alpha_2 \\ \alpha_1 & \alpha_3 \\ \alpha_1 & \alpha_0 \\ \alpha_0 & \alpha_1 \end{bmatrix}.$$

Suppose $\delta_{11} = 0$, $\delta_{12} = 1$, $\delta_{21} = 1$ and $\delta_{22} = 1$. Then $\delta_1 = \alpha_2$ and $\delta_2 = \alpha_3$. Since $\delta_2 = \alpha_2\alpha_1$ and $\delta_2 = \alpha_2^2$, the vector δ is a scalar multiple of the second row vector of M. Therefore it follows that $\epsilon_1 = \alpha_0$, $\epsilon_2 = \alpha_2$ and $\epsilon_3 = \alpha_0$. Hence, $\epsilon_{11} = 0$, $\epsilon_{12} = 0$, $\epsilon_{21} = 0$, $\epsilon_{22} = 1$, $\epsilon_{31} = 0$ and $\epsilon_{32} = 1$.

In the example considered above the number of input variables was small and the Boolean functions required to be synthesized were chosen to be very simple. Therefore the corrector subsystem would probably require more equipment than the encoder subsystem. However, it should be noted that the design of the corrector subsystem is independent of the number of binary input variables and the nature of the original Boolean functions. This design depends only on p and k. Therefore when the number of input variables is large and the Boolean functions required to be synthesized are complicated, the amount of equipment required for the corrector subsystem may be small in comparison to that required for the whole system. This is very desirable, since we assume that the corrector subsystem is highly reliable. The example shows how we can build up the logical design of the corrector subsystem in any general case. However, the author believes that it is possible to build up much more economical corrector subsystems using sequential circuits. Of course, one then pays the penalty of taking a

longer time to correct the errors. Such economical corrector subsystems are discussed in the companion paper,[1] in which minimally redundant reliable systems which correct faults of more than one block are also given.

VII. ACKNOWLEDGMENTS

The author wishes to thank T. H. Crowley, D. B. Armstrong, J. P. Runyon and B. A. Tague for many stimulating and useful discussions.

REFERENCES

1. Armstrong, D. B., this issue, p. 577.
2. Moore, E. F. and Shannon, C. E., Reliable Circuits Using Less Reliable Relays, J. Frank. Inst., **262**, 1956, pp. 191; 281.
3. Tryon, J. G., Redundant Logic Circuitry, U. S. Patent No. 2,942,193.
4. Von Neumann, J., Probabilistic Logics and the Synthesis of Reliable Organisms from Unreliable Components, *Automata Studies*, Annals of Math. Studies, No. 34, Princeton Univ. Press, Princeton, N. J., 1956, pp. 44–98.
5. Lofgren, L., Automata of High Complexity and Methods of Increasing Their Reliability by Redundancy, Inf. & Cont., **1**, 1958, p. 127.
6. Hamming, R. W., Error Detecting and Error Correcting Codes, B.S.T.J., **29**, 1950, p. 147.
7. Bose, R. C., Mathematical Theory of the Symmetrical Factorial Design, Sankhya, **8**, 1947, p. 155.
8. Zierler, N., A Class of Cyclic, Linear, Error-Correcting Codes in p^m Symbols, Group Report 55-19, Lincoln Laboratory.